

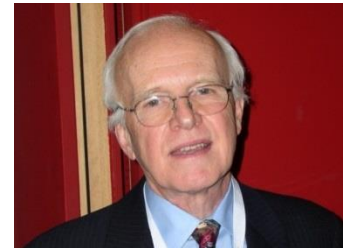
# MÉTRICAS DE SOFTWARE

---

Aseguramiento de la Calidad del software

Prof. Carlos Arias Rodríguez

*“You can’t control what you  
can’t measure”  
Tom DeMarco (1982)*



# ¿Por qué utilizar métricas en el software?

- Facilitar el control de proyectos de desarrollo de software
  - Desviaciones en los atributos de calidad con respecto a lo planeado
  - Desviaciones en calendario y presupuesto con respecto a lo planeado
- Soporte a toma de decisiones y sugerir medidas correctivas/preventivas
  - Acumulación de métricas en el transcurso del tiempo



---

## Métricas deben ser...

Relevantes	Relacionadas con atributos de importancia significativa
Válidas	Realmente miden el atributo requerido
Confiable	Por ejemplo, producen resultados similares cuando se aplican bajo condiciones similares
Exclusión mutua	No miden atributos medidos por otras métricas
Simples y fáciles	La recolección de datos para las métricas no debe requerir demasiados recursos
Inmunes a parcialización	Las personas pueden querer intervenir en los datos por varias razones. Deben definirse procedimientos adecuados para evitar dicha intervención.

---

# Normalizando las métricas

---

*KLOC (Miles de líneas de código).*  
Específicas para un lenguaje de programación determinado

---

*Puntos de función.*  
Medida de los recursos (humanos) requeridos para desarrollar un software, a partir de su funcionalidad

# Métricas del Proceso

1. Calidad del proceso

2. Cumplimiento del cronograma

3. Efectividad en la remoción de errores

4. Productividad

# Métricas del proceso: Calidad

- Cantidad de errores

- Absoluta

- Ponderada

Code	Name	Calculation formula
CED	Code Error Density	$CED = \frac{NCE}{KLOC}$
DED	Development Error Density	$DED = \frac{NDE}{KLOC}$
WCED	Weighted Code Error Density	$WCED = \frac{WCE}{KLOC}$
WDED	Weighted Development Error Density	$WDED = \frac{WDE}{KLOC}$
WCEF	Weighted Code Errors per Function point	$WCEF = \frac{WCE}{NFP}$
WDEF	Weighted Development Errors per Function point	$WDEF = \frac{WDE}{NFP}$

■ NCE = number of code errors detected in the software code by code inspections and testing. Data for this measure are culled from code inspection and testing reports.

■ KLOC = thousands of lines of code.

■ NDE = total number of development (design and code) errors detected in the software development process. Data for this measure are found in the various design and code reviews and testing reports conducted.

■ WCE = weighted code errors detected. The sources of data for this metric are the same as those for NCE.

■ WDE = total weighted development (design and code) errors detected in development of the software. The sources of data for this metric are the same as those for NDE.

■ NFP = number of function points required for development of the software. Sources for the number of function points are professional surveys of the relevant software.

# Métricas del proceso: Cumplimiento de cronograma

Code	Name	Calculation formula
TTO	Time Table Observance	$TTO = \frac{MSOT}{MS}$
ADMC	Average Delay of Milestone Completion	$ADMC = \frac{TCDAM}{MS}$

## Key:

- MSOT = milestones completed on time.
- MS = total number of milestones.
- TCDAM = total Completion Delays (days, weeks, etc.) for All Milestones. To calculate this measure, delays reported for all relevant milestones are summed up. Milestones completed on time or before schedule are considered “0” delays.



# Métricas del proceso: efectividad en remoción de errores

Code	Name	Calculation formula
DERE	Development Errors Removal Effectiveness	$DERE = \frac{NDE}{NDE + NYF}$
DWERE	Development Weighted Errors Removal Effectiveness	$DWERE = \frac{WDE}{WDE + WYF}$

- NYF = number of software failures detected during a year of maintenance service.
- WYF = weighted number of software failures detected during a year of maintenance service.
- NDE = total number of development (design and code) errors detected in the software development process. Data for this measure are found in the various design and code reviews and testing reports conducted.
- WDE = total weighted development (design and code) errors detected in development of the software. The sources of data for this metric are the same as those for NDE.

# Métricas de proceso: productividad

Code	Name	Calculation formula
DevP	Development Productivity	$\text{DevP} = \frac{\text{DevH}}{\text{KLOC}}$
FDevP	Function point Development Productivity	$\text{FDevP} = \frac{\text{DevH}}{\text{NFP}}$
CRe	Code Reuse	$\text{CRe} = \frac{\text{ReKLOC}}{\text{KLOC}}$
DocRe	Documentation Reuse	$\text{DocRe} = \frac{\text{ReDoc}}{\text{NDoc}}$

---

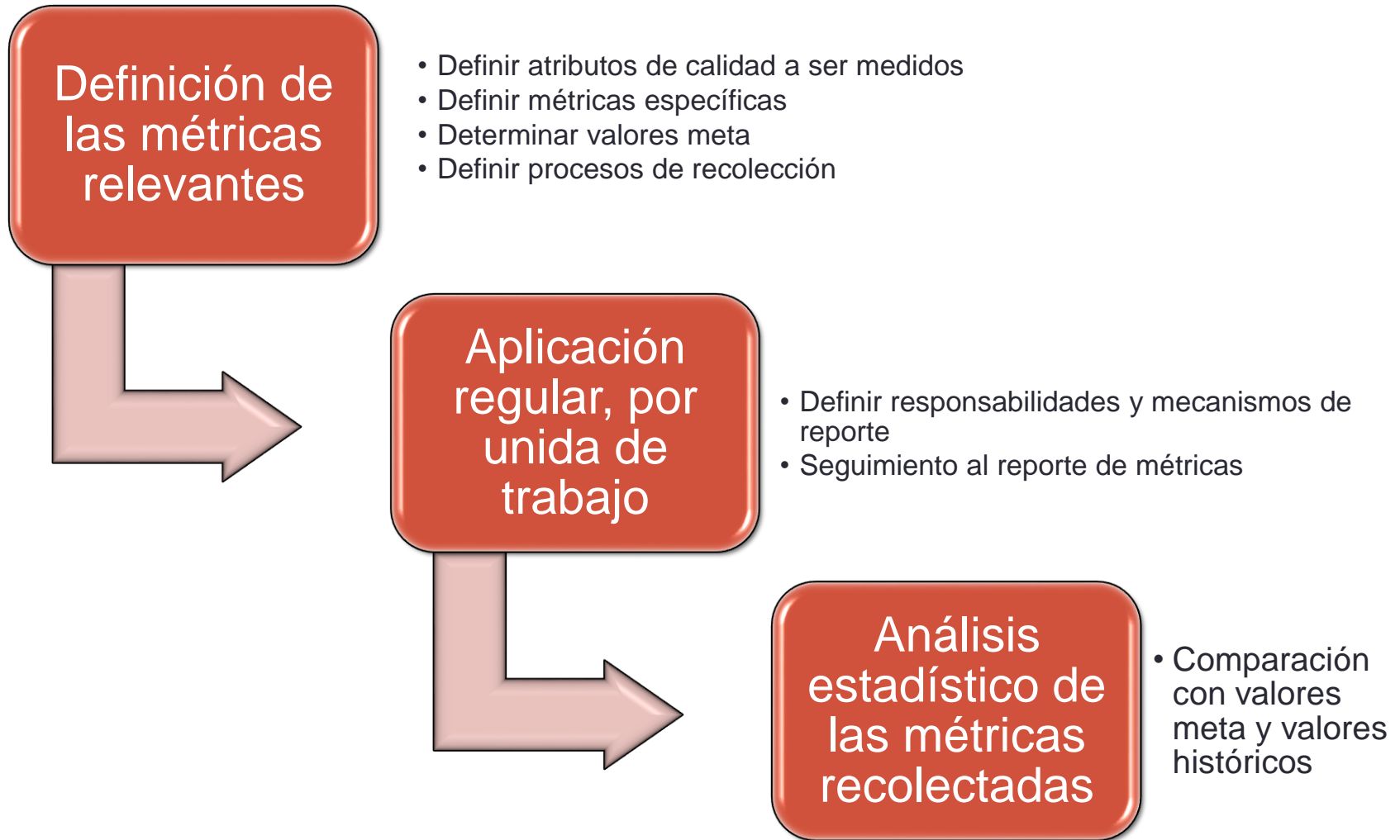
## Key:

- DevH = total working hours invested in the development of the software system.
- ReKLOC = number of thousands of reused lines of code.
- ReDoc = number of reused pages of documentation.
- NDoc = number of pages of documentation.

# Métricas de calidad del producto

- Se refieren al producto durante su fase de operación.
- Evalúan aspectos como:
  - Calidad del servicio *Help Desk* (soporte al usuario)
    - Cantidad de llamadas por año, relativo al tamaño del software
    - Efectividad de los servicios *Help Desk*
  - Calidad de servicios correctivos o de mantenimiento durante la etapa de operación
    - Cantidad y efectividad de servicios correctivos requeridos
    - Disponibilidad del sistema

# Implementación de métricas de calidad



# Limitaciones del uso de métricas

- Limitaciones de presupuesto para instaurar el sistema de métricas
- Factores humanos: oposición de los desarrolladores al control de sus actividades
- Incertidumbre sobre la validez de las métricas



# MÉTRICAS DE SOFTWARE

---

Aseguramiento de la Calidad del software

Prof. Carlos Arias Rodríguez